Object-Oriented Programming Language

09/25/2014

Homework Assignment No. 2

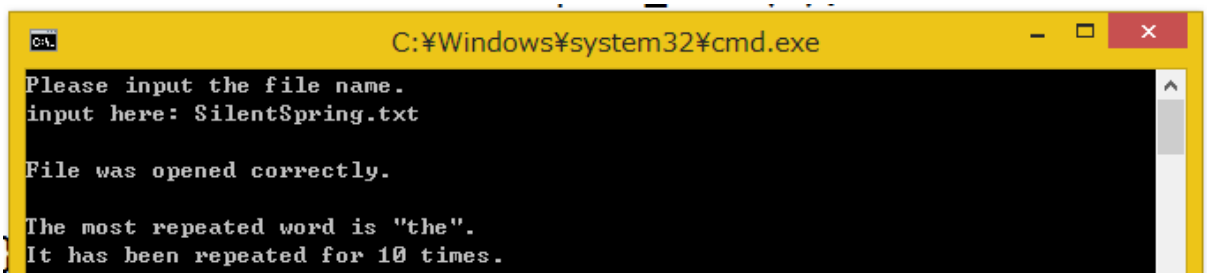
Due 09:00 pm, Wednesday October 1, 2014

Late submission within 24 hours: score*0.9;

Late submission before post of solution: score*0.8 (the solution will usually be posted within a week); no late submission after the post of solution)

(Total 60%)

1. **(30%)** Write a program to read an input file and look for duplicated words. Print the word and the largest number of times a word has occurred. For example, if you take the excerpt of Silent Spring (`SilentSpring.txt`) from the course website, you shall expect the following output:

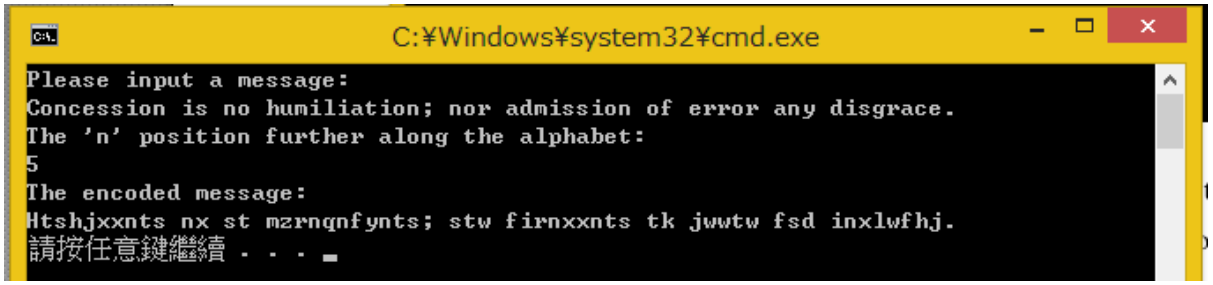


2. **(30%)** A ***Caesar cipher*** encodes a message by replacing each letter of the message with the letter that is n positions further along in the alphabet. For example, if you have a letter `a`, its `2` positions in the alphabet is the letter `c`. Letters at the end of the alphabet wrap around to the beginning. For example, if you have a letter `y`, its `5` positions in the alphabet is the letter `d`. Write a `caesar_cipher` function that takes a message to encode and an integer n as its parameters, and encodes the message using the n-shift.

(The function header)

```
string caesar_cipher(const string& input, const int shift);
```

(A sample run)

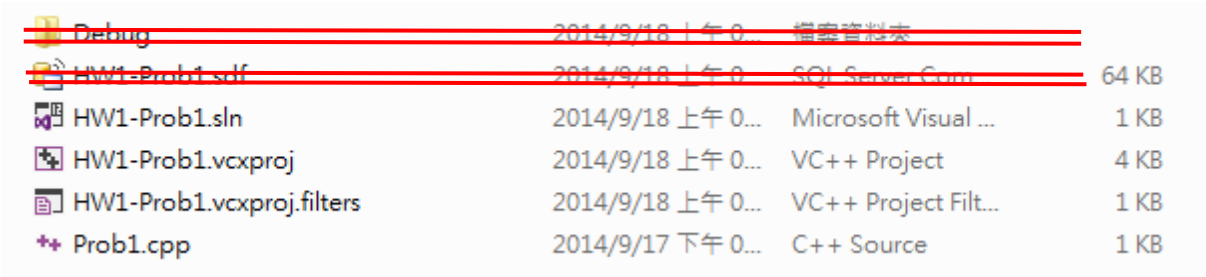


Object-Oriented Programming Language

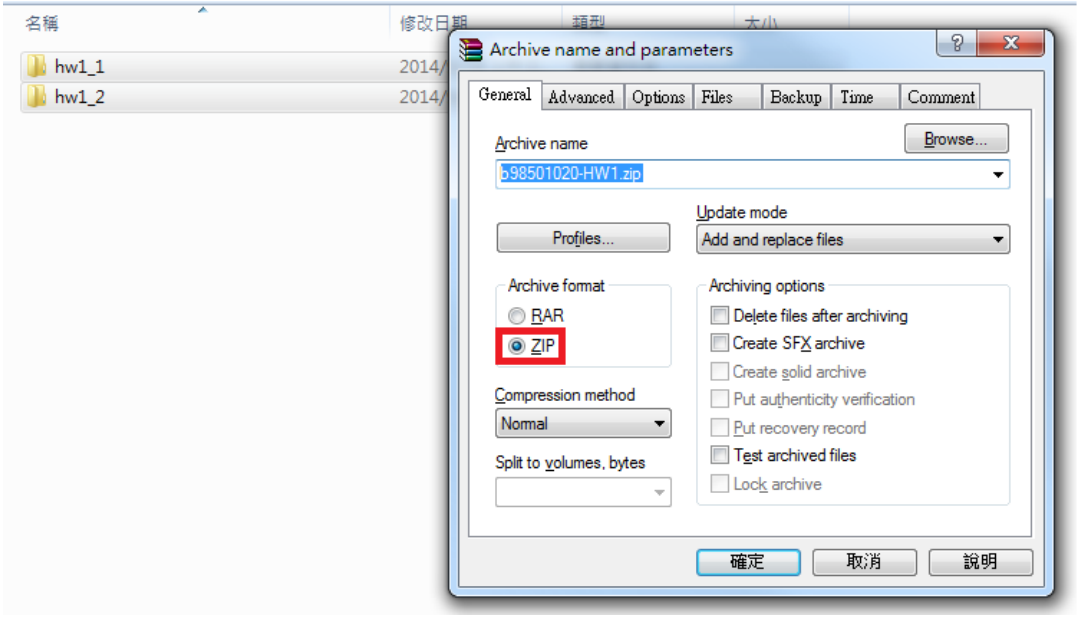
09/25/2014

HW Submission Procedure (請仔細閱讀):

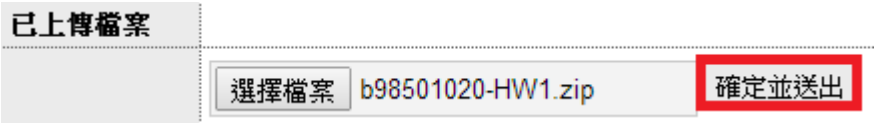
1. You should create an independent Visual Studio 2013 project for each problem. You should submit your project folder, which includes your source codes (header and cpp files), input/output data and Visual Studio 2013 project files. To save space, please delete Debug folder and sdf before submitting your files.
   繳交時請以專案資料夾區分每一道題目，資料夾中應包含該題中的每個標頭檔、cpp 檔、所讀取的資料與輸出的結果與 Visual Studio 2013 相關專案檔。繳交前請刪除 Debug 資料夾與 sdf 檔案，以節省空間。



2. You should zip all the files and use your student id to name the zip file (e.g., b98501020-HW1.zip).
   將所有檔案及資料夾收納在以學號命名的 zip 壓縮檔中。(例如：b98501020-HW1.zip)



3. Submit your HW directly through the course website.
   請直接透過課程網站繳交作業。



HW Grading Policy:

1. You should consider about exception handling, e.g. error input, file opening fail, etc.

請注意所有例外狀況的處理，例如：錯誤的符號字串輸入、檔案開啟失敗等。

2. The coding style includes your output format.
   輸出資料的格式將納入格式評分。
3. If your code is not compilable, your score in this problem is zero (including coding style).
   若程式無法編譯，則該題以零分計算。(包含格式分數)
4. Your program will be tested with other data which is not the same as provided samples.
   除了題目所提供的範例測試資料以外，作業程式碼將以額外的測試資料進行測試。
5. If tricky situations occur, the grade depends on Prof. Chen or TA's judgment.
   假如有特殊情況發生，則依據陳俊杉教授以及助教們的判斷給分。

- Coding Style (20%): 編碼格式分數
    1. format
       整體形式與輸出資料的格式
    2. comments
       註解
    3. readability
       可讀性
    4. variables naming
       變數命名方式
    5. typesetting
       型別設定

- Functionality (80%): 功能性分數
    1. run-time performance:
       執行時的表現
        1) samples not passed -> x
           範例測資錯誤 => 此部分零分
        2) samples passed but some tests failed -> partial
           範例測資通過但是部分測資失敗 => 部份給分
        3) samples and tests all passed
           範例測資與所有測資通過 => 此部分滿分
    3. excellent method++
       綜合以上，又以能展現解決問題的巧思尤佳。